ANALYSIS OF SECURE DISTRIBUTED DEDUPLICATION SECRET DATA IN CLOUD COMPUTING

R.Dhivya, L.Jeevitha,
Department of Computer Science & Computer Applications,
Padmavani Arts and Science College for Women, Salem.

ABSTRACT

Data deduplication is one of the key data compression techniques to eliminate duplicate copies of repetitive data and has been widely used in cloud storage to reduce disk space and save bandwidth. To protect the confidentiality of sensitive data during deduplication, the converged encryption technology has been proposed to encrypt the data before outsourcing. In order to better protect the data security, this paper makes the first attempt to address the issue of authorized deduplication officially. Unlike traditional deduplication systems, the duplicate privileges of the users are considered further apart from the data themselves. We also present some, in this proposed new deduplication designs that support an authorized duplicate check in a hybrid cloud architecture. The safety analysis shows that our scheme is safe with regard to the definitions defined in the proposed safety model. As proof of the concept, we analysis implement a prototype of our proposed duplicate test system and carry out test-test experiments with our prototype. The proposed duplicate check scheme has a minimal outlay compared to normal operation.

Keywords: *Deduplication, Hybrid Cloud Architecture, bandwidth, encryption, prototype*

1. INTRODUCTION

Personal Health Care (PHC) is an emerging patient-oriented model of health information exchange that is often outsourced to be stored with a third party such as cloud providers. However, there was far-reaching privacy as personal health information could be exposed to these third parties and unauthorized parties. In order to ensure the control of patients through access to their own PHCs, it is a promising method to encrypt the PHCs before outsourcing.

However, issues such as the risks of privacy exposure, scalability in key management,
flexible access, and efficient user revocation are key challenges for obtaining fine-grained, cryptographically enforced data access control. In this article, we propose a novel patient centric framework and a set of mechanisms for data access control for PHCs stored in semi-trusted servers.

2. RELATED WORK

MihirBellare[1] has proposed theCloud storage service providers such as Dropbox, Mozy, and others perform deduplication to save space by only storing one copy of each file uploaded. Should clients conventionally encrypt their files, however, savings are lost. Message-locked encryption (the most prominent manifestation of which is convergent encryption) resolves this tension. However it is inherently subject to brute-force attacks that can recover files falling into a known set.

SriramKeelveedhi [2] has proposed the architecture that provides secure deduplicated storage resisting brute-force attacks, and realize it in a system called DupLESS. In DupLESS, clients encrypt under message-based keys obtained from a key-server via an oblivious PRF protocol.

Thomas Ristenpart [3] has proposed the enables clients to store encrypted data with an existing service, have the service perform deduplication on their behalf, and yet achieves strong confidentiality guarantees. We show that encryption for deduplicated storage can achieve performance and space savings close to that of using the storage service with plaintext data.

Jin Li [4] has proposed theData deduplication is a technique for eliminating duplicate copies of data, and has been widely used in cloud storage to reduce storage space and upload bandwidth. Promising as it is, an arising challenge is to perform secure deduplication in cloud storage.

Xiaofeng Chen[5] has proposed the convergent encryption has been extensively adopted for secure deduplication, a critical issue of making convergent encryption practical is to efficiently and reliably manage a huge number of convergent keys. This paper makes the first attempt to formally address the problem of achieving efficient and reliable key management in secure deduplication.

Mingqiang Li[6] has proposed the first introduce a baseline approach in which each user holds an independent master key for encrypting the convergent keys and outsourcing them to the cloud. However, such a baseline key management scheme generates an enormous number of keys with the increasing number of users and requires users to dedicatedly protect the master keys.

Jingwei Li[7] has proposed theDekey, a new construction in which users do not need to manage any keys on their own but instead securely distribute the convergent key shares across multiple servers.

Patrick P.C. Lee[8] has proposed the proof of concept, we implement Dekey using the Ramp secret sharing scheme and demonstrate that Dekey incurs limited overhead in realistic environments.

Paul Anderson[9] has proposed theMany people now store large quantities of personal and corporate data on laptops or home computers. These often have poor or intermittent connectivity, and are vulnerable to theft or hardware failure.

Le Zhang[10] has proposed theConventional backup solutions are not well suited to this environment, and backup regimes are frequently inadequate. This paper describes an algorithm which takes advantage of the data which is common between users to increase the speed of backups, and reduce the storage requirements. This algorithm supports client-end per-user encryption which is necessary for confidential personal data.

Chun-Ho Ng[11] has proposed theDeduplication is known to effectively eliminate duplicates, yet it introduces fragmentation that degrades read performance.

Patrick P. C. Lee[12] has proposed theRevDedup, a deduplication system that optimizes reads to the latest backups of virtual machine (VM) images using reverse deduplication. In contrast with conventional deduplication that removes duplicates from new data, RevDedup removes duplicates from old data, thereby shifting fragmentation to old data while keeping the layout of new data as sequential as possible. We evaluate our RevDedup prototype using a 12-week span of real-world VM image snapshots of 160 users. We show that RevDedup achieves high deduplication efficiency, high backup throughput, and high read throughput.

3. SYSTEM DESIGN

The system design describes an extended schema to support stronger security by encrypting the

file with differential authorization keys. In this way, the users can not perform the duplicate check without appropriate privileges. In addition, such unauthorized users can not decrypt the ciphertext, even collide with the S-CSP. The safety analysis shows that our system is safe with regard to the definitions defined in the proposed safety model.

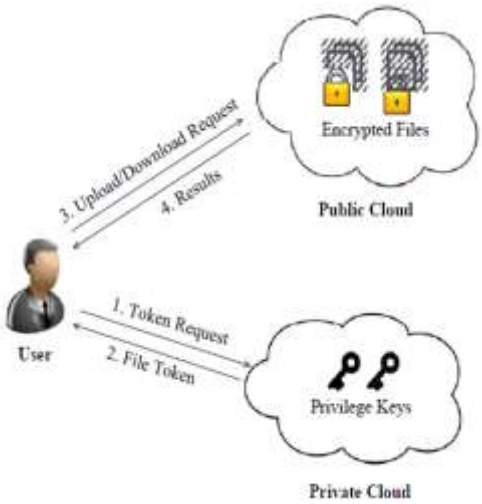

Figure 3.1 System Design

4. PROPOSED METHOD

In this paper, we enhance our system in security. Specifically, we present an advanced scheme to support stronger security by encrypting the file with differential privilege keys. In this way, the users without corresponding privileges cannot perform the duplicate check. Furthermore, such unauthorized users cannot decrypt the cipher text even collude with the S-CSP. Security analysis demonstrates that our system is secure in terms of the definitions specified in the proposed security model.We propose a new cloud computing paradigm, data protection as a service (DPaaS)is a suite of security primitives offered by a cloud platform,which enforces data security and privacy and offers evidenceof privacy to data owners, even in the presence ofpotentially compromised or malicious applications.Such as secure data using encryption, logging, key management.

4.1 Cloud Computing

Cloud computing is providing dynamically scalable and often virtualized resources as services over the Internet users do not need to know know-how or control over the technology infrastructure in the "cloud" that supports them. Cloud computing is a major change in how we store information and run applications. Instead of placing hosts and data on a single desktop computer, everything is hosted in the "cloud," a collection of computers and servers accessed over the Internet.

4.2 Trusted Platform Module

Trusted Platform Module (TPM) is both the name of a published specification that specifies a secure crypto processor, the cryptographics that protect information, and the general name of the implementations of this specification, often referred to as a "TPM chip" or "TPM Security "Can store device." The TPM specification is the work of Trusted Computing Group.

Disk Encryption is a technology that protects information by converting it into unreadable code that can not be easily decrypted by unauthorized persons. The disk encryption uses hard disk encryption software or hardware to encrypt every bit of data that runs on a volume or volume volume. The disk encryption prevents unauthorized access to the data storage. The term "full disk encryption" (or entire disk encryption) is often used to mean that everything is encrypted on a disk, including programs that can encrypt bootable operating system partitions. But they still have to leave the master boot record (MBR) and thus a portion of the hard drive is not encrypted. However, there are hardware-based full-disk encryption systems that can really encrypt the entire bootdisk, including the MBR.

4.3 Third Party Auditor

In this module Auditor examines all user data and checks data as well as changed data. Auditor directly views all user data without key. Admin has given the authorization to the auditor. After auditing data to save the cloud.

4.4 User Module

Users store large amount of data on clouds and access data with secure keys. Secure key provided admin after encrypting data. Encrypt the data with TPM. Users store data by auditor, view and verify data, and also changed data. Users re-enter data at this time admin, provided the message to user changes only data.

5. PROPOSED BASED ON QUERY ALGORITHM.

Input: Pairwise similarity values S ¼ fsijji ¼ 1; . . .;N; j ¼ 1; . . .;Ng where sij is the similarity between sentences i and j. Number of clusters, C.

Output: Cluster membership values fpmi

ji ¼ 1; . . . ;

N;m ¼ 1; . . . ; Cg

1. // INITIALIZATION
2. // initialize and normalize membership values
3. for i ¼ 1 to N
4. for m ¼ 1 to C
5. pmi

¼ rnd // random number on [0, 1]

6. end for
7. for m ¼ 1 to C
8. pmi ¼ pmi = PC j¼1 pj i // normalize
9. end for
10. end for
11. for m ¼ 1 to C
12. _m ¼ 1=C // equal priors
13. end for
14. repeat until convergence
15. // EXPECTATION STEP
16. for m ¼ 1 to C
17. // create weighted affinity matrix for cluster m
18. for i ¼ 1 to N
19. for j ¼ 1 to N
20. wmij ¼ sij _ pmi _ pmj
21. end for
22. end for
23. // calculate Page Rank scores for cluster m
24. repeat until convergence
25. PRmi ¼ ð1 _ dÞ þ d _PN j¼1 wmjiðPRmj = PN k¼1 wmjkÞ
26. end repeat
27. // assign PageRank scores to likelihoods
28. lmi ¼ PRmi
29. end for
30. // calculate new cluster membership values
31. for i ¼ 1 to N
32. for m ¼ 1 to C
33. pmi ¼ ð_m _ lmi Þ= PC j¼1ð_j _ ljiÞ
34. end for
35. end for
36. // MAXIMIZATION STEP
37. // Update mixing coefficients
38. for m ¼ 1 to C
39. _m ¼ 1N PN i¼1 pmi
40. end for
41. end repeat

6. RESULTS AND DISCUSSION

Cloud computing is providing dynamically scalable and often virtualized resources as services over the Internet users do not need to know know-how or control over the technology infrastructure in the "cloud" that supports them. Cloud computing is a major change in how we store information and run applications. Instead of placing hosts and data on a single desktop computer, everything is hosted in the

"cloud," a collection of computers and servers accessed over the Internet.

Figure 7.1 Login Page

Figure 7.2 Home Page

Figure 7.3 Data Entry Page

Figure 7.4 Account Creation Page

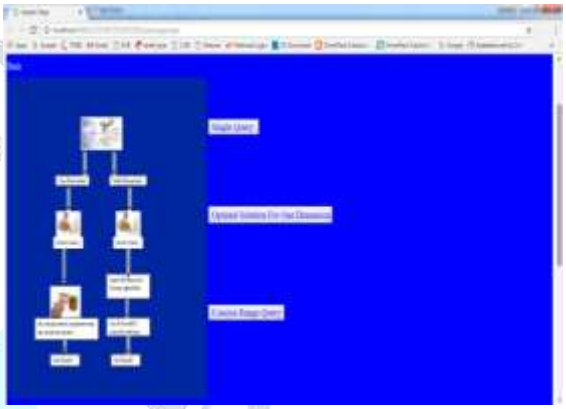

Figure 7.5 Query Searching Page

Figure 7.6 Single Query Search Page

Figure 7.7 Multi Query Search Page

Figure 7.8 Result Page

The above images illustrate how an account
Created using account creation page
Using multi query search page data deduplication identified and displayed in result page

7. CONCLISION AND FUTURE WORK

As private data is moved online, the need to secure it properly becomes more and more urgent. The good news is that the same forces that focus data in enormous data centers are also helping to leverage collective security competency more effectively. Adding protection to a single cloud platform can immediately benefit from hundreds of thousands of applications and hundreds of millions of users. While we have concentrated here on a certain, though popular and privately sensitive, class of applications, many other applications also need solutions.

It excludes the security problems that may arise during the practical introduction of the present model. It also increases national security. It saves the memory by de-duplication of the data and thus provides us with sufficient memory. It provides the private companies with an authorization and protects the confidentiality of the important data.

8. REFERENCES

[1] M. Abdalla, M. Bellare, and P. Rogaway, "The oracle diffiehellman assumptions and an analysis of DHIES," in Proc. Conf. Topics Cryptol., vol. 2020, 2001, pp. 143–158.

[2] B. Adida, "Helios: Web-based open-audit voting," in Proc. 17thUSENIX Security Symp., 2008, pp. 335–348.

[3] R. J. Anderson, Security Engineering : A Guide to Building Dependable Distributed Systems, 2nd ed. New York, NY, USA: Wiley 2008.

[4] A. Antipa, D. Brown, A. Menezes, R. Struik, and S. Vanstone, "Validation of elliptic curve public keys," in Proc. 6th Int. Workshop Practice Theory Public Key Cryptography Public Key Cryptography, 2003, pp. 211–223.

[5] S. Bauer and N. B. Priyantha, "Secure data deletion for linux file systems," in Proc. 10th USENIX Security, 2001, pp. 153–164.

[6] C, Burton, C. Culnane, J. A. Heather, P. Y. A. Ryan, S. Schneider, T. Srinivasan, V. Teague, R. Wen, and Z. Xia, "Using pret a voter in victorian state elections," in Proc. Electron. Voting Technol./Workshop Electron. Voting, 2012, pp. 1–16.

[7] C. Cachin, K. Haralambiev, H. C. Hsiao, and A. Sorniotti, "Policybased secure deletion," in Proc. ACM Conf. Comput. Commun. Security, 2013, pp. 259–270.

[8] R. Geambasu, T. Kohno, A. Levy, and H. M. Levy, "Vanish: Increasing data privacy with self-destructing data," in Proc. 18th Conf. USENIX Security Symp., 2009, pp. 299–316.

[9] S. Garfinkel and A. Shelat, "Remembrance of data Passed: A study of disk sanitization practices," IEEE Security Privacy, vol. 1, no. pp. 17–27, Jan. 2003.

[10] P. Gutmann, "Data remanence in semiconductor devices," in Proc. 10th Conf. USENIX Security Symp., 2001, pp. 39–54.

[11] F. Hao, M. Kreeger, B. Randell, D. Clarke, S. Shahandashti, and P. Lee, "Every vote counts: Ensuring integrity in large-scale electronic voting," USENIX J. Election Technol. Syst., vol. 2, no. pp. 1–25, 2014.

[12] N. Joukov, H. Papaxenopoulos, and E. Zadok, "Secure deletion myths, issues, and solutions," in Proc. 2nd ACM Workshop Storage Security Survivability, 2006, pp. 61–66.

[13] M. Kallahalla, E. Riedel, R. Swaminathan, Q. Wang, and K. Fu, "Plutus: Scalable secure file sharing on untrusted storage," Proc. 2nd USENIX Conf. File Storage Technol., 2003, pp. 29–41.

[14] R. Kissel, M. Scholl, S. Skolochenko, and X. Li, "Guidelines for media sanitization," NIST Special Publication, vol. 800–88, 2006.

[15] T. Kohno, A. Stubblefield, A. D. Rubin, and D. S. Wallach, "Analysis of an electronic voting system," in Proc. 25th IEEE Symp. Security Privacy, May, 2004, pp. 27–40.

[16] J. Lee, S. Yi, J. Y. Heo, H. Park, S. Y. Shin, and Y. K. Cho, "An efficient secure deletion scheme for flash file systems," J. Inf. Sci. Eng.vol. 26, pp. 27–38, 2010.